



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO



Aplicación para comunicaciones en red
3CM17



Practica 04

"Servidor HTTP"

Integrantes:

- ♥ Bocanegra Heziquio Yestlanezi
- ♥ Martinez Cruz José Antonio

Profesor: Moreno Cervantes Axel Ernesto

Índice

Índice de imágenes.....	3
INTRODUCCION.....	4
DESARROLLO	5
Método POST	6
Postman.....	7
CONCLUSIONES	9
Bocanegra Heziquio Yestlanezi	9
Martínez Cruz José Antonio.....	9



Índice de imágenes

Imagen 1 formulario GET	5
Imagen 2 contenido disponible	6
Imagen 3 formulario POST	6
Imagen 4 cliente conectado.....	7
Imagen 5 métodos	7
Imagen 6 DELETE	7
Imagen 7 dirección si modificación	8
Imagen 8 cambio aplicado.....	8



INTRODUCCION

En la programación de computadoras, un grupo de subprocesos es un patrón de diseño de software para lograr la concurrencia de ejecución en un programa de computadora. A menudo también llamado modelo de trabajadores replicados o modelo de trabajadores, un grupo de subprocesos mantiene varios subprocesos esperando que el programa de supervisión asigne tareas para la ejecución concurrente. Al mantener un grupo de subprocesos, el modelo aumenta el rendimiento y evita la latencia en la ejecución debido a la frecuente creación y destrucción de subprocesos para tareas de corta duración. El número de subprocesos disponibles se ajusta a los recursos informáticos disponibles para el programa, como una cola de tareas paralelas después de la finalización de la ejecución.



El tamaño de un grupo de subprocesos es el número de subprocesos que se mantienen en reserva para ejecutar tareas. Suele ser un parámetro ajustable de la aplicación, ajustado para optimizar el rendimiento del programa. Decidir el tamaño óptimo del grupo de subprocesos es crucial para optimizar el rendimiento. Se ha sugerido la técnica de análisis de grupo de subprocesos (HTA) basada en la hipérbola para determinar el tamaño óptimo del grupo de subprocesos para el proceso de indexación basado en la nube en función de la carga de trabajo y el ancho de banda disponibles.

Mime.

Un tipo MIME es una etiqueta utilizada para identificar un tipo de datos. Se utiliza para que el software pueda saber cómo manejar los datos. Tiene el mismo propósito en Internet que las extensiones de archivo en Microsoft Windows.

Entonces, si un servidor dice "Esto es texto / html", el cliente puede ir "Ah, este es un documento HTML, puedo representar internamente", mientras que si el servidor dice "Esto es aplicación / pdf", el cliente puede ir "Ah", Necesito iniciar el complemento FoxIt PDF Reader que el usuario ha instalado y que se ha registrado como el controlador de aplicaciones/pdf.

Los encontrará más comúnmente en los encabezados de los mensajes HTTP (para describir el contenido con el que responde un servidor HTTP o el formato de los datos que se PUBLICAN en una solicitud) y en encabezados de correo electrónico (para describir el formato del mensaje y archivos adjuntos).

DESARROLLO


En esta práctica se hicieron peticiones como clientes a un servidor, que actúa como un apache, nos brinda un servicio con el método POST y GET.

Como primer lugar veamos que son el método GET:

Cuando un usuario rellena un formulario en una página web los datos hay que enviarlos de alguna manera. Vamos a considerar las dos formas de envío de datos posibles: usando el método POST

En el ejemplo anterior la acción que se ejecutará cuando el usuario pulse el botón “Enviar datos” (submit) será el envío de los datos a la url especificada usando el [método GET](#)

Veamos el aspecto de un formulario cualquiera para hacernos una idea general.



Contenido disponible

Formulario GET

Nombre:

Dirección:

Teléfono:

Comentarios:

Imagen 1 formulario GET

Este formulario consta de varios campos que al usuario se le solicitan como Nombre, Dirección, Teléfono y Comentarios.

La diferencia entre los métodos get y post radica en la forma de enviar los datos a la página cuando se pulsa el botón “Enviar”. Mientras que el método GET envía los datos usando la URL, el método POST los envía de forma que no podemos verlos (en un segundo plano u "ocultos" al usuario).

Al llenarlo nos aparece lo siguiente:

Contenido disponible

Practica3Servidor.rar

Practica3Micros.pdf

Peach.jpg

Practica01.docx

Samus.jpg

Rosalina.png

VideoPerfecto.mp4

Practica 16.c

shaq.jpg



Imagen 2 contenido disponible

Método POST

Hemos visto el resultado de un envío por el método GET. En el caso de un envío de datos usando el método POST, aunque estos datos también serán enviados (de una forma que podemos denominar “oculta”), no los podremos ver en la URL. Para poder recuperar los valores de los campos en el caso de un envío con el método POST necesitaríamos otras herramientas (por ejemplo, valernos del lenguaje PHP para recuperar el valor de esos campos).

Así como cuenta con la parte del formulario POST que podemos observar en la siguiente imagen.

Formulario POST

Nombre:

Dirección:

Teléfono:

Comentarios:

Enviar datos

Borrar datos

Imagen 3 formulario POST

Postman

Ahora para consultar los métodos solicitados realizaremos la conexión como se muestra en la siguiente imagen 4

```
-----> Cliente Conectado desde: /127.0.0.1
Por el puerto: 55098
Datos: PUT /contenido/sofia.jpg HTTP/1.1

Respuesta PUT:
HTTP/1.1 200 OK
Date: Wed Jun 14 20:53:45 CDT 2023
Server: Server/1.0
Content-Type: text/html
```



Imagen 4 cliente conectado

Una vez realizada la conexión exitosamente podemos realizar los siguientes métodos como se muestra en la imagen 5

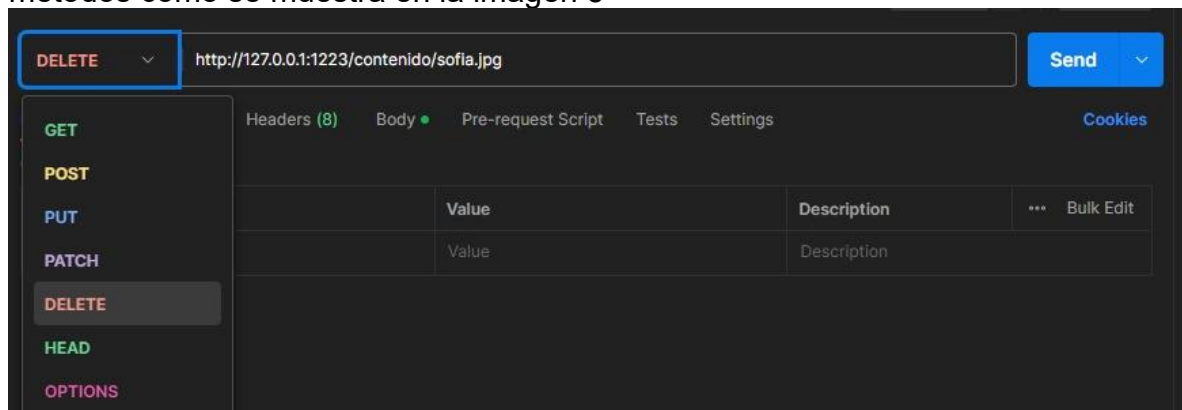


Imagen 5 métodos

Para este ejemplo utilizaremos DELETE, cabe mencionar que el principal error al utilizarlo es que la dirección de donde se están tomando los archivos no es la correcta, así que una vez asegurados de que la dirección es la correcta, seleccionamos el método DELETE seguido de la imagen o archivo que deseamos eliminar, el cual debe estar escrito exactamente igual que como aparece en la carpeta.

En nuestro caso podemos observar que después de la dirección utilizamos **/sofia.jpg** este será el archivo a eliminar, una vez dando en "Send" apareciendo en la consola como se muestra en la imagen 6.


```
Datos: DELETE /contenido/sofia.jpg HTTP/1.1

Por el puerto: 55422
contenido/sofia.jpg
Datos: GET /contenido.html HTTP/1.1

-----> Archivo contenido/sofia.jpg eliminado exitosamente
```

Imagen 6 DELETE

El mensaje anterior nos estará indicando que ya se elimino como se puede observar en la imagen 7 y 8.



Nombre	Fecha de modificación	Tipo	Tamaño
Otros	07/06/2023 09:42 p. m.	Carpeta de archivos	
Peach.jpg	13/04/2023 02:29 p. m.	Archivo JPG	18 KB
Practica01.docx	28/04/2023 09:56 p. m.	Documento de Mi...	6,784 KB
Practica03_Equipo7_MartinezCruzJoseAn...	25/04/2023 06:07 p. m.	Documento Adob...	814 KB
Practica3Servidor.zip	07/06/2023 09:33 p. m.	Archivo WinRAR Z...	499 KB
practica16.c	11/05/2023 01:14 p. m.	Archivo C	8 KB
ros.jpg	08/06/2023 09:52 a. m.	Archivo JPG	1 KB
Samus.jpg	07/06/2023 09:49 p. m.	Archivo JPG	239 KB
shaq.jpg	12/11/2020 05:36 p. m.	Archivo JPG	36 KB
sofia.jpg	14/06/2023 07:53 p. m.	Archivo JPG	1 KB
VideoPerfecto.mp4	06/03/2023 07:40 p. m.	Archivo MP4	1,203 KB



Imagen 7 dirección si modificación

en la imagen anterior podemos observar que el archivo sofia.jpg se encuentra ahí, debido a que es la carpeta original con los archivos originales antes de aplicar las modificaciones.

Otros	07/06/2023 09:42 p. m.	Carpeta de archivos	
Peach.jpg	13/04/2023 02:29 p. m.	Archivo JPG	18 KB
Practica01.docx	28/04/2023 09:56 p. m.	Documento de Mi...	6,784 KB
Practica03_Equipo7_MartinezCruzJoseAn...	25/04/2023 06:07 p. m.	Documento Adob...	814 KB
Practica3Servidor.zip	07/06/2023 09:33 p. m.	Archivo WinRAR Z...	499 KB
practica16.c	11/05/2023 01:14 p. m.	Archivo C	8 KB
ros.jpg	08/06/2023 09:52 a. m.	Archivo JPG	1 KB
Samus.jpg	07/06/2023 09:49 p. m.	Archivo JPG	239 KB
shaq.jpg	12/11/2020 05:36 p. m.	Archivo JPG	36 KB
VideoPerfecto.mp4	06/03/2023 07:40 p. m.	Archivo MP4	1,203 KB

Imagen 8 cambio aplicado

En la imagen 8 ya podemos observar que después del cambio aplicado, ya no aparece en la carpeta el archivo sofia.jpg lo que nos indica que el método funciono de manera correcta.

CONCLUSIONES

Bocanegra Heziquio Yestlanezi

Para la realización de esta práctica, los métodos aplicados fueron diferentes, ya que tuvimos que utilizar otro programa del cual no teníamos conocimiento para aplicar los métodos especificados por el profesor, aunque después de ser explicado a detalle como funcionaba el programa, tuvimos algunos problemas con el funcionamiento de nuestro código, debido a que no estábamos colocando la dirección correcta de donde este tomaría la información para aplicar los métodos solicitados, pero después de un tiempo nos percatamos de ello y pudimos solucionarlo, siendo así que la practica se realizo en su totalidad con éxito, adquiriendo los conocimientos esperados e implementando nuevas técnicas para los servidores HTTP.



Martínez Cruz José Antonio

Con esta práctica se logró entender el funcionamiento del protocolo HTTP y cómo podemos identificar cada uno de sus encabezados con sus principales características. Aunque existieron inconvenientes con el uso de los encabezados DELETE y PUT, se logró solucionar estos detalles permitiendo hacer la subida de un archivo y poder eliminar un archivo recién creado o ya creado utilizando la herramienta postman que espero en un futuro pueda utilizarla de manera más ágil.

